Alex Allmont

Software Engineer

info@alexallmont.com
+44 (0)7795 614374

40 Slaymaker Close
Headington
Oxford
OX3 8EP

Profile

With over 20 years of experience C++, combined with C, Python, Rust and Ruby, I have worked with scientists, engineers and artists to develop robust technology from concept to production. I deliver well-tested, maintainable and performant code, using object and data-oriented, and functional methodologies as appropriate. C++ includes 11/14/17/20, STL, Boost, OpenGL and Qt; Linux, Windows, OSX and embedded.

Oxford Instruments – Systems and Embedded C++ Developer – July 2023 to present

Oxford Instruments produce refrigeration and superconducting magnet systems for in material science and quantum computing. I worked on embedded drivers, new database design using PostgreSQL and TimescaleDB, and improved our developer tooling to integrate C++ debugging with our Python test system.

OxeHealth Ltd – Systems C++, Python and Rust Developer – January 2022 to July 2023

OxeHealth use deep learning video technology to improve patient feedback for clinicians. I worked in C++ 17, Python and Rust using a docker-based Linux environment built on CMake, GitLab AWS, SQL and Parquet.

C++/Python bridge: Pybind11 C++ bindings to integrate legacy system into latest Python research projects.

Rust AWS tooling: Data processing tools to reduce AWS S3 costs.

Research team tooling: GUI features, algorithmic optimisation, docker composed video debug tools.

OxSonics Ltd – C++ Developer Contract – December 2020 to February 2021

SonoTran is a revolutionary cancer treatment that uses ultrasound to localise drug deployment. I worked on GTest and Qt unit tests in preparation for IEC 62304 medical testing device standards.

Oxford Nanopore Technology Ltd – Senior C++ and Python Developer – August 2017 to April 2020

VolTRAX is a portable device for automating biological sample preparation. The core CMake C++ 14 project uses Boost and Protobuf to communicate with an Electron front-end. My personal work included:

C++ development: Core C++ API consolidating comms, telemetry and mocking for testing. Image processing algorithms for tracking droplet splits and merges. Smoothed-particle hydrodynamic fluid simulation.

Microfluidic Python API: Created 'DropTRAX' generative 2D animation system to allow scientists to rapidly deveop lab protocols for dispensing, moving, mixing and extracting fluids. The API is an animation graph using image processing to maintain state, developed in close collaboration with researchers.

Diagnostic UI: Web interface for live interaction of fluid on the device, along with stats and debug viewers.

Hardware: Embedded USB devices for automating tests, mentoring engineers in firmware code.

Production: MongoDB queries to report customer usage and analyse hardware faults.

NaturalMotion Ltd – Software Developer and Dev Process Lead – April 2006 to June 2017

Morpheme is an animation system for interactive game animation. I worked on the GUI for creating and debugging animation networks using C++, wxWidgets and Win32 API. My personal work includes:

Lua bindings: C++ metatemplate API for scripted UI. Used in dev, support and by customers.

3D scene graph: Node graph API for rapid development of scenes used in Morpheme Connect.

Asset management: Responsive Win32 multithreaded code for tracking and interacting with assets.

Workflow: Ruby Qt tools for SVN workflow, CI, docs generation and code review. One-to-one assistance.

Teamwork: Acting Head of Delivery. Coding standards. Mentoring. Geometric algebra lecture.

Alex Allmont

Software Engineer

info@alexallmont.com
+44 (0)7795 614374

40 Slaymaker Close
Headington
Oxford
OX3 8EP

ArtiCAD Ltd – Head of Programming – June 2001 to April 2006

ArtiCAD allows rapid prototyping of kitchens and bathrooms on-site with customers. I started as the sole developer to take on the legacy C++ OWL codebase, and oversee the team as it grew.

Script API: Lua bindings for rapid development of bespoke customer requirements.

Preview tools: Streamlined OpenGL UI workflow to view components whilst editing parameters.

Autodesk integration: Importers and exporters for Autodesk DXF and 3D Studio using MaxSDK.

Sony Computer Entertainment Europe – Games Developer – February 2000 to June 2001

Working with SCEE's Team Soho I developed game tech in C for PlayStation 1 and in C++ for PlayStation 2, requiring rapid development of robust and maintainable code, including:

Game UI: Interactive UI for This is Football 2 on PS1 in collaboration with artists.

Manager AI: Simulated team manager system for This is Football 2.

Traffic simulation: AI system for traffic for The Getaway on PS2, using route and curve solving.

Applied Communications Inc. – Programmer Analyst – February 1998 to February 2000

ACI develop financial software used in cash machines and servers. I worked on server-side software using a Pascal/C hybrid called TAL, and client-side UI in COBOL. Code was carefully reviewed and tested to ensure it was delivered to specification. Key work included:

Cryptographic integration: Hardware secure interface between Lloyds and TSB systems.

Chip card interchange: Enhancing existing system to support EuroPay chip card transactions.

Client support: On-site work with Natwest and LloydsTSB configuring and supporting systems.

Education

Oxford Brookes University: MA Distinction in Contemporary Art and Music – 2009 to 2011.

Oxford University: Mathematics, programming and psychology part-time courses – 2009, 2010, 2018, 2022.

Open University: MST121: Using Mathematics – 2004.

University of Plymouth: Computer Systems and Networks (incomplete) – 1995 to 1997.

Gordano Comprehensive School: 1987 to 1994, including A-level maths and physics.

Interests

I enjoy combining ideas with hands-on skills to experiment with music, maths, electronics and art projects. My musical work has been featured in Wired Magazine and shown at Music Tech Fest, which has led me to speak at several events. I have presented at UCL, Festival of the Spoken Nerd and Maths Jam, and have run workshops for museums and universities.

Whenever possible I take on classes to catch up on subjects, most recently on the psychology of play and sculpture. I like to learn by experimentation, presently exploring Rust for eye exercise software, Python for generative music concepts, and OpenSCAD and OnShape for 3D design. More broadly, I enjoy playing guitar and experimenting with synthesizers.